

A Collaborative Decision Environment for UAV Operations

Matthew V. D'Ortenzio*, Francis Y. Enomoto†, and Sandra L. Johan‡
NASA Ames Research Center, Moffett Field, CA, 94035, U.S.A.

NASA is developing Intelligent Mission Management (IMM) technology for science missions employing long endurance unmanned aerial vehicles (UAV's). The IMM ground-based component is the Collaborative Decision Environment (CDE), a ground system that provides the Mission/Science team with situational awareness, collaboration, and decision-making tools. The CDE is used for pre-flight planning, mission monitoring, and visualization of acquired data. It integrates external data products used for planning and executing a mission, such as weather, satellite data products, and topographic maps by leveraging established and emerging Open Geospatial Consortium (OGC) standards to acquire external data products via the Internet, and an industry standard geographic information system (GIS) toolkit for visualization. As a Science/Mission team may be geographically dispersed, the CDE is capable of providing access to remote users across wide area networks using Web Services technology. A prototype CDE is being developed for an instrument checkout flight on a manned aircraft in the fall of 2005, in preparation for a full deployment in support of the US Forest Service and NASA Ames Western States Fire Mission in 2006.

Nomenclature

BLM	= Bureau of Land Management
CDE	= Collaborative Decision Environment
CIP	= Collaborative Information Portal
COLA	= Common Outer Loop Architecture
CORBA	= Common Object Request Broker Architecture
FAA	= Federal Aviation Administration
GIS	= Geographic Information System
HALE	= high altitude long endurance
http/https	= hypertext transfer protocol/secure hypertext transfer protocol
IMM	= Intelligent Mission Management
MODIS	= Moderate Resolution Imaging Spectroradiometer
NIFC	= National Interagency Fire Center
OGC	= Open Geospatial Consortium
SPS	= Sensor Planning Service
UAV	= unmanned aerial vehicle
USFS	= United States Forest Service
VIS-IR-TIR	= Visible-Infrared-Thermal Infrared
WFS	= Web Feature Service
WMS	= Web Map Service

I. Introduction

THIS paper discusses the motivation for, and the design of the Collaborative Decision Environment (CDE), a software system developed to support Unmanned Aerial Vehicle (UAV) operations. The CDE is the ground-based component of the Intelligent Mission Management (IMM) project, whose aim is to increase the efficiency and effectiveness of UAV operations through an infusion of proven autonomy technologies. A brief discussion of the IMM project will follow, highlighting the CDE's role in the overall program vision. The CDE's concept of

* Computer Engineer, Intelligent Systems Division, M/S 269-1, Moffett Field CA 94035.

† Aerospace Engineer, Intelligent Systems Division, M/S 269-1, Moffett Field CA 94035, AIAA 004887.

‡ Computer Engineer, Intelligent Systems Division, M/S 269-1, Moffett Field CA 94035.

operations, requirements and design will be discussed, focusing on technology choices made during the course of the design. As the CDE has its heritage in NASA's Collaborative Information Portal (CIP) project, CIP's features will be mentioned in order to illuminate the evolution of the technology, and further demonstrate the extensibility inherent in the underlying architecture.

II. Intelligent Mission Management

Intelligent Mission Management is a sub-project of the Autonomous Robust Avionics Project (AuRA) within NASA's Vehicle Systems Program.¹ The IMM sub-project is working to extend mission-level autonomy for UAV systems that can be broadly tasked by humans who are not highly trained vehicle operators. The goal of IMM is to reduce operator workload and increase mission effectiveness for HALE aircraft, specifically targeting sustained operations (100+ days) and next generation planetary aircraft. Operator workload can be reduced using fault tolerant software for tactical maneuvering, intelligent flight management, and automated reasoning (dynamic re-planning) in both nominal and emergency conditions. Mission effectiveness can be increased through the utilization of a collaborative decision environment for mission-level decision support, automated data products, and sensor planning services. The program is focused on the development of two core products: Common Outer-Loop Architecture (COLA), which addresses the operator workload, and the Collaborative Decision Environment (CDE), which addresses mission effectiveness.

The CDE is to provide a distributed, graphical application for mission planning and decision support, mission monitoring and interaction with the on-vehicle autonomy during the execution of a mission; the CDE will also display geo-referenced data products from numerous automated sources, including the mission payload.

IMM is closely partnered with Principal Investigators and engineers from the Earth Science and land management communities. These relationships provide strong, real-world mission pull for the technologies developed, as well as access to appropriate sensor payloads developed by stakeholders. One such mission is a 24-hour wildfire reconnaissance, surveillance and mapping flight, to be conducted over the western United States. This mission has been developed under the NASA/U.S. Forest Service Wildfire Research and Applications Partnership (WRAP) and will make use of IMM decision support systems during planned 2006 flight.² The COLA will be demonstrated for a similar western states fire mission on the newly acquired Predator-B vehicle in 2007.

III. CDE Concept of Operations and Reference Mission

While the vision for the CDE is one that has it supporting a broad range of missions, during the concept and requirement definition phase, it was useful to select a particular mission, and use its operational scenarios and goals to drive the initial design. As part of the aforementioned partnership between the IMM project and Earth Science community at NASA Ames, the Western States Fire Mission (WSFM) was chosen as our reference mission concept. Requirements definition was further driven by domain expert interviews with U.S. Forest Service staff.

The WSFM is collaborative effort between NASA Ames and U.S. Forest service to demonstrate enhancements to the UAV as a platform for fire imaging capabilities. The overarching goal of the WSFM is to demonstrate methodologies to collect and distribute real-time, geo-registered, multi-spectral wildfire image data from a long-duration mission-capable UAV operating at altitudes between 20K and 45K feet. The specific capability to be demonstrated is a 24-hour flight over various fire events throughout the western United States. The IR data collected on-board is to be delivered, in a timely and usable fashion, to a wildfire Incident Command (IC) team actively engaged at a fire site.

The mission will be flown using a General Atomics ALTAIR UAV, operating out of Grey Butte, CA. A 12-channel multi-spectral sensor, fine tuned for fire imaging will be flown as the primary payload along with the ALTAIR's Skyball video system, which will be used for targeting and alignment of the UAV platform and the sensor during approach to a targeted fire. The delivered data will consist of a 3-band, VIS-IR-TIR image of the target areas in GeoTIFF format. The delivery of a GeoTIFF product is made possible because the geo-registration, terrain corrections and other appropriate corrections will all be performed on-board. This is a major advancement in fire imaging operational systems, and greatly reduces the time delay between acquisition and delivery to the end user. Additionally, the sensor will be re-configurable in flight, allowing selection of any three of the twelve bands.

Such a mission has several areas where the CDE can be effective in ensuring mission effectiveness. Mission-level decision support, data product access and visualization, and situational awareness are some of these areas that the CDE could address, and so requirements were written to formalize the CDE's role in the WSFM in each of them. Figure X shows the resulting layout of the mission and indicates which systems the CDE interfaces with.

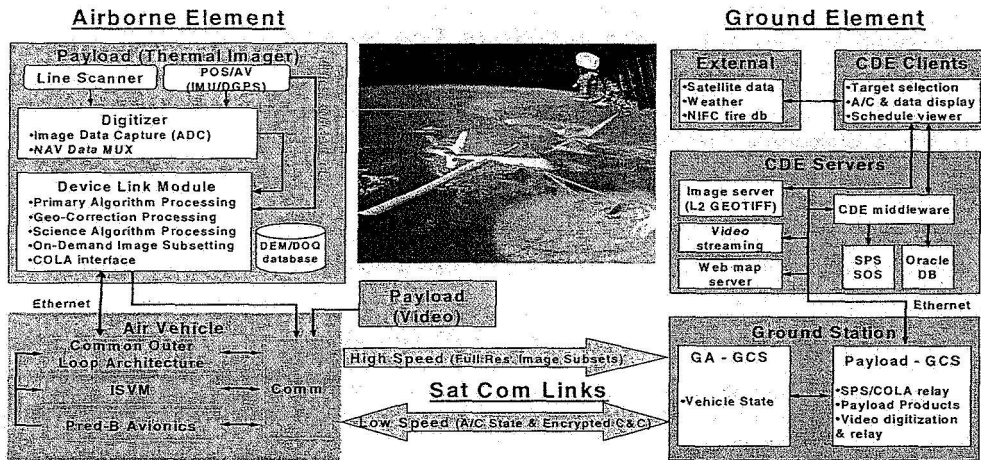


Figure 1. WSFM Block Diagram. High level view of system components for the WSFM. "Airborne element" refers to the ALTAIR UAV, and "Payload" refers to the 12-channel imager. The "Ground Station" is the General Atomics provided ground support trailer with pilot station and communication and control capability.

1. Decision Support

A multitude of factors will go into the flight planning activity for the WSFM. The goal of the mission is, of course, to image active wildfires. However, during fire season, at any one time there may be dozens of fires burning, which necessitates a prioritization of fires to be imaged. Choosing the fires that will yield the most benefit to the Incident Command team will require coordination with NIFC, as well as data from a variety of sources. The CDE should be able to access the USFS's database of active fires, as well as data from satellite resources such as MODIS fire detections. The data returned will provide an initial set of targets for the flight planner and mission manager to choose from.

Efficiency during flight should be maximized to obtain the largest amount of useful data, while at the same time maintaining safe operation. Because the UAV will be flying in the National Airspace System (NAS), the flight must stay strictly within its FAA authorized airspace. Therefore, display of the mission's "Certificate of Authorization" (COA) boundaries alongside the proposed flight path is a critical function of the CDE. Weather is also a major consideration, not only from an aircraft safety perspective, but also for the success of the data collection. Weather can interfere with both the operation of the payload instrument and the quality of the data that is returned. The CDE must also afford the flight planning personnel the ability to access and display pertinent weather information alongside the active fires that are being considered for observation.

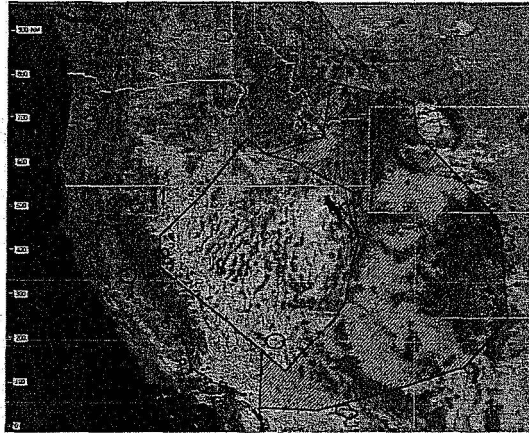


Figure 2. WSFM COA Boundary. Shaded area in purple represents the desired flight area; blue represents a back-up area. The COA boundaries are encoded as a GIS "feature" then overlaid on a base image.

2. Mission Data Access

The CDE will serve as the primary data portal for the WSFM staff and collaborators. Accordingly, data products gathered by the payload system should be available for display and download by the CDE client. These data include the GeoTIFF scene images returned by the payload system as well as a vector file containing the computed active fire perimeter in ESRI Shapefile format. Because both data products mentioned are geo-referenced, the preferred method of visualizing them is in a tool with GIS-type functionality, which allows a user to overlay such data on a base map. Such a tool should also be able to incorporate any pertinent geospatial data that enhances analysis capability such as roads, structure locations, vegetation information, and bodies of water.

3. Situational awareness

The ALTAIR UAV operator station can accommodate approximately five individuals; namely the pilot, the payload operators and the mission managers. Yet the mission stakeholders encompass a much larger group that will not necessarily reside at the flight operations facility. Providing a high level of situational awareness to all mission staff, both at Grey Butte, and elsewhere is critical for mission success. The CDE will provide users with up-to-date aircraft position and status information, as well as payload information, and will also need to notify users when new data products have been delivered from the sensor.

Beyond knowledge of the present, situational awareness also requires knowledge of the progression of the mission and what has occurred up to the present. Therefore, the CDE should have the ability to display the mission's overall timeline, and indicate where in the timeline the mission currently is. Since offsite users won't be actively involved in the piloting of the mission, knowing the intentions of the aircraft, and what it is planning to do next, is as important as knowing where it is at any given point in time.

Additionally, in the event of an anomaly or other occurrence this is only apparent to the onsite staff; users should have a mechanism to broadcast a message to make the rest of the staff aware of the incident. Simple messages like "data delivery will be delayed" would be extremely helpful to other staff waiting for data products. The opposite should also be possible. If an offsite user observes something in the aircraft status information or in the acquired data that requires attention from other staff, they should also be able to broadcast their findings.

IV. CDE Design

The design of a prototype CDE was completed using the WSFM as a "concept of operations" and groundwork was laid for development of a mission-worthy product by 2006. Using the WSFM concept several key CDE design drivers were readily identified. Firstly, the CDE had to be a network application. Given that the system will have users in a number of different locations, all with the need to access image and status data, and be kept abreast of the current situation, makes this clear. The need for inter-user communication, and for the publication and notification of data arrival, furthers the point. Additionally, in such a network environment, the system also has to enforce a user authentication and authorization scheme so that access to data and services is partitioned and protected.

From the user interface perspective, it was obvious that a GIS-style component was needed, that could not only visualize a variety of geospatial data, but could access it over a network. Because of the variety in anticipated users, this component would also have to be highly configurable. Because, a variety of products and protocols do exist for these types of application, a thorough trade study was performed to evaluate which would be the most valuable.

The resultant CDE design to meet these demands was an n-tier, pure Java application, utilizing Enterprise Java Beans and Web Services technology on the server side and a Java Swing on the client side. This solution was very much in the spirit of the previously successful Collaborative Information Portal, which had similar but not identical requirements and concept of operations. Yet the CDE design realized several key advancements on the CIP concept including the addition of a GIS toolkit embedded in the client application, the adoption of several OGC protocols for accessing GIS content and driving mission execution, and variations on the back-end architecture to enable near real-time operations.

A. Design Elements

1. Collaborative Information Portal

The Collaborative Information Portal was enterprise software developed jointly by the NASA Ames Research Center and the Jet Propulsion Laboratory (JPL) for the highly successful Mars Exploration Rover (MER) project.³ Used throughout the mission, CIP provided a series of useful tools that assisted project scientists and engineers in performing their daily tasks such as schedule tracking, announcement broadcasting, and easy data product access.

Initially designed as an aid to mission management, CIP enjoyed usage from a broad range of mission staff, from members of rover operations team, to the publicity and outreach coordinators.

CIP was unique among the software used on the mission in that it was a distributed system that could be accessed on or off site. This allowed users to keep track of mission progress even when away from JPL. Simple, but essential tools, such as a multi-time zone (including Mars zones) clock display, staff and event schedule viewers and a data product tracker gave remote staff members a level of situational awareness otherwise unavailable.

The CIP project was able to provide a rich client experience to both remote and local users through the development of a 3-tiered network application, which included a "thick" Java client tier and a service-oriented middleware tier utilizing web services and Enterprise Java Beans. The choice to develop a "thick" client, as opposed to a "thin", or web-browser based client was an important one as it freed client programmers from the limitations imposed by a browser environment, and the choice to use Java guaranteed that the client could be used on all desktop computing platforms used by the mission staff (Windows, Mac, Linux, Solaris).

CIP's Web Services based architecture and client tools were largely applicable to the user requirements of the CDE. The service-oriented approach simplified the addition of new services needed to satisfy the CDE demands, and the client interface, with its pluggable component model, was also easily expandable. While several of the CIP tools could be used across domains, and incorporated directly into the CDE, others were tightly coupled with the unique MER data model, and so were not included in the CDE implementation.

2. GIS Toolkit

For the WSFM, providing a geographic context in which to visualize the returned data products is highly important. Without such an environment, any data products that the sensor system could generate would provide little usable information in operational activities. Incident Command teams tend to agree with this idea, as evidenced by the steady increase in strategic usage of GIS services on modern incidents. For example, in recent years Pacific Northwest Team 3 has consistently requested a team of GIS analysts accompany them on the large "type 1" incidents that they deal with. When on scene the GIS teams are able to produce useful mapping products derived from field observations, IR sensor flights and county or state provided data. On fire incidents the ability to manipulate various "layers" of data in one geographic view is highly effective in tracking the progression of the fire, predicting its growth, and identifying risks to firefighters.

There are a variety of GIS tools available today, although ESRI products are considered the standard. ESRI Inc. is one of the main providers of commercial GIS software, and because it was one of the pioneers in the field, its products are widely used throughout the GIS industry and the government; including USFS and BLM. The recognition of the reality drove the selection of ESRI's MapObjects Java Version as the GIS toolkit to be used in the CDE. Because most ESRI products contain similar features, CDE users who have used such products should find its interface familiar.

Because MapObjects is Java software, it is embeddable in other Java applications and can be run on all main Java platforms. Choosing MapObjects allowed CDE development to leverage much of the CIP client application while providing users with a familiar and powerful set of GIS functionality that would otherwise take significant effort to duplicate. However, other products, including several strong open-source toolkits do exist, and have equivalent features. Subsequent versions of the CDE may see the migration to a different product provided that the toolkit is embeddable and extensible.

The GIS features required to fulfill the CDE requirements include the following:

- Panning and zooming of the display
- Z-order modification of data and image layers in the display
- Ability to import image layers from ArcIMS Image and OGC Web Map servers
- Ability to import data layers from ArcIMS Feature and OGC Web Feature servers
- Display of data in a wide range of map projections (geographic, UTM, etc.)
- Ability to perform spatial queries for features contained in the imported layers

It is of note that both of the open source products that were evaluated featured native support for OGC Web Map Service⁵ and OGC Web Feature Service⁶ data access. WMS and WFS are open standards for accessing geo-referenced image and data content over the internet. Many data providers, including USGS, NOAA, BLM and USFS have implemented such services to serve data to a wider variety of users. At the time of this writing WMS and WFS support was not included in ESRI MapObjects Java Edition. As a result, CDE development time was spent adding the required code to add support for these important services.

B. CDE Architecture

Like CIP, CDE is a three-tiered enterprise system. Users run local copies of the client application that uses the Internet to access shared data. On the server side, middleware software handles simultaneous data requests from the client applications, and securely accesses “backend” data repositories needed to fulfill those requests, mainly the CDE Oracle databases containing metadata, schedules, and the message archive. See Figure 1. The CDE follows a software system model known as a service-oriented architecture (SOA). A SOA consists of a loosely coupled collection of services, where each service is a well-defined, self-contained function, independent of other services. The services communicate with each other and with the client applications through a set of protocols known as web services.⁶

The SOA concept is demonstrated clearly in the CDE’s method of authenticating users. To perform a “log in”, a CDE client first connects to the *User Management Service* running in the middleware server. Via a Web service invocation, the client makes a request for an *Access Token*, receives the token (assuming supplied credentials are valid), and then disconnects. The *User Management Service* keeps track of all active user sessions and valid tokens. The client application is then able to access other middleware services, such as those for data or schedule access, by presenting the token when making requests. Upon receiving such requests the other services make a Web service call to the *User Management Service* to validate the token. Such a scheme enforces clear delegation of function loose coupling.

Web services communicate using a textual XML-based industry-standard protocol known as SOAP[§]. Service requests and responses are actually small XML documents passed between the client and server. CDE client and server transmit these documents securely using HTTPS. The XML documents that are transmitted, and Web services in general, are language independent, as the web services standard defines a finite set of XML-based data types to be used. Therefore, any programming language that has library routines to communicate via SOAP and to convert between native data types and the XML data types can use web services. Both the CDE client application and the server-side Web service implementations are written entirely in Java, however the server could be accessed by any Web service enabled client, such as one written in C++ or VB.NET. See Figure 3.

The selection of Web services technology offers other important features. Web services use the ubiquitous hypertext transfer protocol (http) for transport, and transmission is performed over the common ports for that protocol, namely port 80 for standard, and port 443 for secure. This greatly simplifies network security configuration since the clients communicate with the server using the same ports and the same protocol as web browsers. Special firewall configuration, common when using CORBA or other middleware, is not needed since these ports are broadly considered as “standard”. Additionally, web services do not require persistent connections. A client connects to the middleware server, makes a request, receives the response, and disconnects. Clients installed on mobile platforms (laptops, etc.) are then able to cope more readily with intermittent network connectivity.

C. Client Software

The CDE application is a “thick client” desktop application, as opposed to a “thin client” application that runs within a web browser. A thick client makes better use of the user’s local computer and provides better interactivity and responsiveness. The client application is written using the widely available Java platform and graphical user interface components from its Java Foundation Classes (“Swing”).

Figure 7 shows the component-based approach for the client tier. Each client tool is a CDE Component object, and a Service

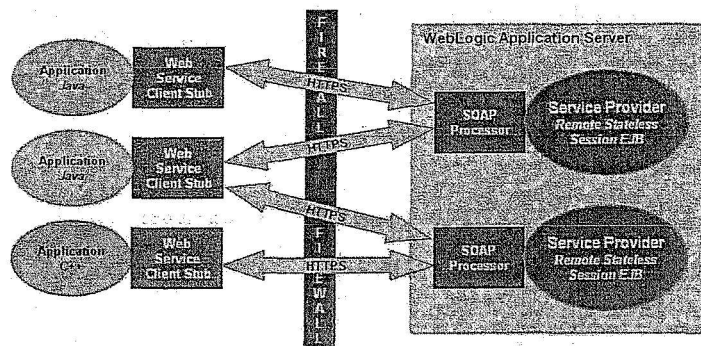


Figure 3. Web Services and Service Provider Beans in the Middleware

[§] SOAP originally stood for Simple Object Access Protocol. Now the acronym supposedly doesn’t stand for anything, although some claim it ought to stand for Service-Oriented Architecture Protocol.

Manager object supported one or more CDE Component objects. Each Service Manager object manages the connections to a particular remote middleware service by using a Web Services Client Stub. For example, the clock components use the Time Service Manager object, which managed the connections to the middleware's time service. The Web Services Client Stubs is responsible for the conversion between the clients' native data types and the XML data types.

Because the CIP client, and subsequently the CDE client, uses the component-based approach, integration of new CDE components into the client framework was a straightforward process, as was reuse of existing CIP components. What follows is a description of how the developed components work together to achieve the desired CDE functionality:

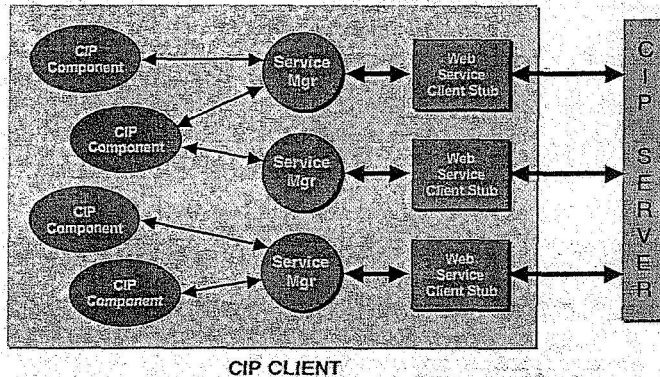


Figure 4. The Component-Based Client Application Architecture with Web Services

1. Mission Planning:

One goal of the CDE and IMM is to involve the end-user of the acquired data product in the pre-flight decision making process, while shielding them from the operational details of the specific aircraft. Project scientists and Fire Command staff are concerned with the phenomenon to be observed, and where best to observe it, not whether the UAV has enough fuel to get home, or has a tight enough turning radius to get from one location to another. Therefore, the interface into the flight planning process needs to be intuitive from a mission planning perspective, and focus on the mission level goals.

The Planning Tool within the CDE fulfills this role by providing a mechanism for the user to select targets for investigation as well as annotating those targets with constraint information. Targets can either be specified as a *point* defined by a geographic location (e.g. lat./long.), a *line* specified by two locations, or as a bounded *area* defined by three or more locations. Available constraints include a target's priority, order of observation, and time of day. Once complete, the set of targets, which now comprises an *experiment plan*, is given to the COLA Planning Executive. With the target information, the executive attempts to create a flight plan that satisfies as many of the experiment plan's goals and constraints as possible. Unlike a traditional flight planner, the user is not selecting waypoints for the aircraft to follow. Instead he is making the mission-level decision about which areas to investigate, while the autonomy makes the operational decisions about how and where to fly the aircraft.

As shown in Figure X the Planning Tool uses the GIS software as its base component. With a GIS tool as an underlying component, the user is able to leverage a variety of external data sources by importing data and image layers according to their needs, and perform the selection within a geographic context. Fire commanders could visualize any of the numerous GIS data products common to fire operations including IR maps, fire line maps, and asset positions. Science mission planners could access weather imagery or satellite overpass data to coordinate concurrent data collection. Having this type of extensible interface allows the user to create their own customized view, and therefore makes the CDE applicable to various types of users across a variety of applications.

Like other CDE components, the Planning Tool is able to operate in a collaborative manner. While users are able to produce experiment plans for "off-line" use and "what-if" analysis, the experiment plan that the aircraft will be tasked with is an aggregation of the inputs from a larger number of users.

Comment: Does this include the SPS?

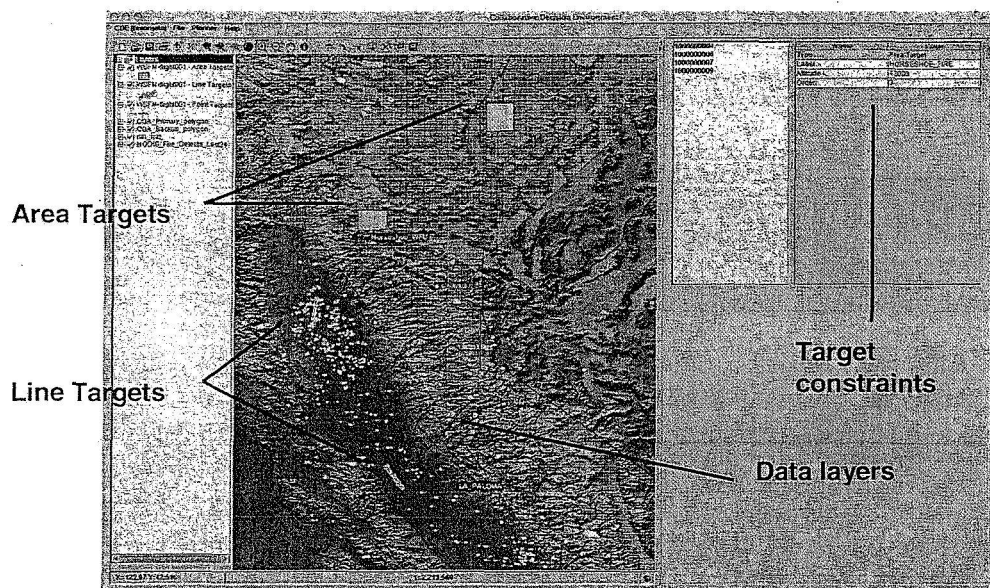


Figure 5. CDE Planner Tool. Screen shot of Planner Tool showing line and area target selection (light blue). The GIS viewer is in the center configured with a base map and a data layer containing recent MODIS fire detections (yellow). The table of contents for the GIS viewer is on the left, and the target constraint and annotation area is on the right. (Screen shot from Mac OS)

2. Mission Monitoring:

Another goal of the CDE and IMM is to allow users to monitor mission progress. As mentioned earlier, mission teams are typically distributed, so providing mechanisms to promote situational awareness are fundamental to mission success. Monitoring is one such mechanism. It allows team members to know what is happening with respect to the mission, regardless of their geographic location. For example, all team members should be able to quickly see where the UAV is currently, and where it is going next.

The CDE has created two new tools and leveraged several existing tools from CIP to support mission monitoring. The new tools are the Overview Map and the Monitor. Both tools provide information in a GIS display; however, each has a different intended use. The Overview Map is intended to provide users with a strategic or wide-angle view of the mission. To achieve the strategic effect, the Overview Map's display is initially configured to capture the entire mission area, and all layers are configured for that scale. For mission such as the WSFM, a coarse fire map is also initially displayed. Its placement within the client user interface further enunciates its role as a strategic tool. The Overview Map resides in the upper portion of the CDE, a typically visible area; the user can thus readily monitor the location of the aircraft with respect to the mission area extent. See Figure X.

The Monitor tool provides users with a more general GIS display with a tactical and "zoomed-in" view of the area in which the aircraft is flying and attempting to image. Whereas the Overview Map would show the user that the aircraft is over Northern California, the Monitor tool display would show over what town, street, or landmark was being over flown. Because the tool resides in the lower tabbed portion of the CDE display, it has more screen space available, which allows the rendering of more detailed maps and data layers. Like all tools in this area, the Monitor may be detached, or "torn off" into its own separate window, and later re-docked as desired.

Both tools use a map viewer as described earlier; however, each is typically configured differently. Users can also customize the maps to meet their specific needs. If desired, they can save their configurations on the CDE server as well as revert back to the system default map configurations.

Other CDE monitoring tools are carried over from CIP. The Schedule Viewer enables mission monitoring by displaying staff and event schedules, enabling users to discover when events occur, who is working when and where, and what roles they need to fill that day. During the MER mission, having easy access to current schedules was especially helpful as regularly scheduled events, while constant in a Mars time zone, drifted later from day to

day relative to Earth time. This generally won't be a problem on most UAV missions; however, having mission staff in several different Earth time zones can cause problems without proper synchronization. The Schedule Viewer ameliorates the situation by providing the same up-to-date schedule information to all users, whether local or remote.

The content of the schedules is left to the discretion of the mission managers. However, for UAV applications we anticipate the need for display of key events, such as "time-en-route" to the next target location, time until next data product availability, or local overpass time of pertinent satellite resources.

Other tools leveraged from CIP include an Event Horizon display for tracking selected events, a configurable stack of Clocks that can display time in most Earth time zones, and a broadcast announcement tool for sending messages to other CDE users.

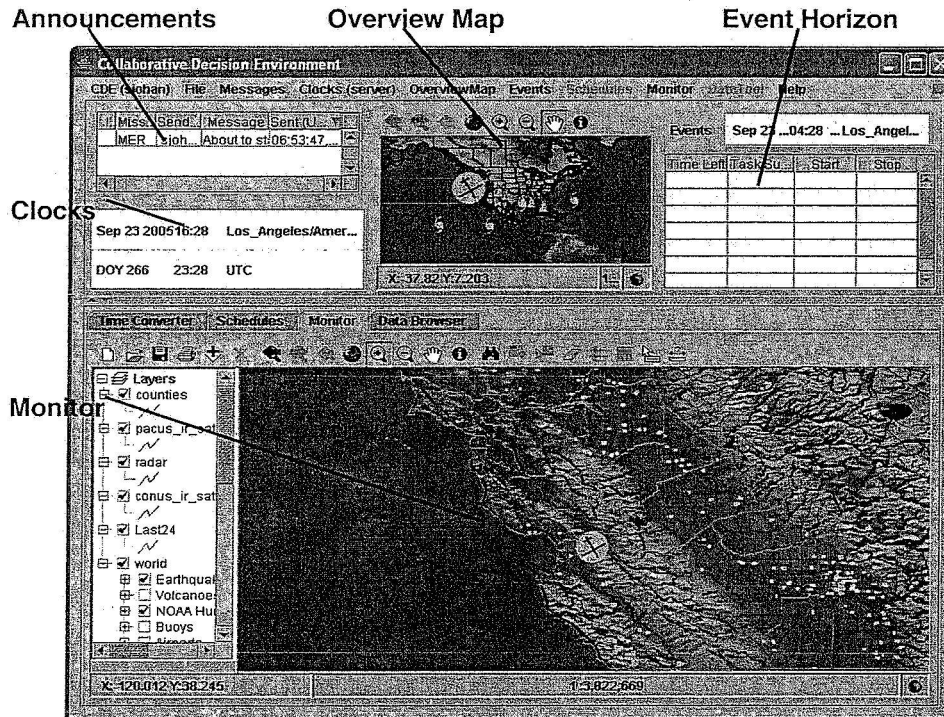


Figure 6. CDE Client showing Monitor Tool. Screen shot showing the Monitor tool in the lower area of the screen and the Overview map in upper, or "manager's area". Aircraft position is shown by a configurable icon, shown as yellow circle in this instance.(Screen shot from Windows XP).

V. Conclusion

Because of its strong heritage in proven software (CIP), there is much confidence in the CDE's success in a mission environment; both in terms of reliability and impact. Progress towards the support of the WSFM in 2006 has been substantial in the past year, although there are still some features to be added. Most pressing is the ability to integrate a view of the ALTAIR's video stream, and better map production and printing facilities. Yet, because the vision for IMM and the CDE extends beyond the WSFM, much long term work and planning remains. The CIP/CDE infrastructure, and client software is based on five year old technology. An honest technology evaluation should be performed to verify that what is being used currently is the best choice for meeting the CDE's

requirement. Striking the correct balance between the software's ability to adapt to different missions, with its ability to prove useful for a particular mission is a constant challenge; however, the adaptation of CIP to CDE has shown it can be done.

Acknowledgments

The authors would like to acknowledge the following for their support and dedication the CDE development process: Louise Chan (SAIC), Tarang Patel (SAIC), Quit Nguyen (QSS Group), and Rajkumar Thirumalanambi (QSS Group). The authors would also like to thank our Western States Fire Mission collaborators for their invaluable assistance in deriving CDE user requirements: Vince Ambrosia (CSU Monterey Bay), Everett Hinkley (USFS), Stephen Wegener (BAER Institute), and Tom Zajkowski (USFS). A special thanks also goes to Randy Herrin (BLM) and the entire Pacific Northwest Incident Management Team 3 for allowing us to get a first hand account of a wildfire incident.

References

- ¹Sullivan, D., Totah, J., Wegner, S., Enomoto, F. Y., Frost, C., Kaneshige, J., and Frank, J., "Sensor Directed Mission Management for Unmanned Aerial Vehicles," *SPIE Fourth International Asia-Pacific Environmental Remote Sensing Symposium: Remote Sensing of the Atmosphere, Ocean, Environment, and Space, Honolulu, HI, November 2004*.
- ²Ambrosia, V. G., Schoenung, S. M., Wegener, S. S., and Enomoto, F. Y., "The 24-Hour UAV Western States Fire Mission: Sensor and Intelligent Management Systems," *AIAA Infotech@Aerospace, Advancing Contemporary Aerospace Technologies and Their Integration Conference, Meeting Papers on Disc (CD-ROM), AIAA, Arlington, VA, 2005*.
- ³Mak, R., Walton J., Keely, L., Heher, D., Chan, L., "A Reliable Service-Oriented Architecture for NASA's Mars Exploration Rover Mission", *IEEE Internet Computing*.
- ⁴De La Beaujardiniere, Jeff, "OpenGIS Web Map Service (WMS) Implementation Specification", www.opengeospatial.org/specs, Document 04-024, v1.3, 2004.
- ⁵Vretanos, Peter, "OpenGIS Web Feature Service (WFS) Implementation Specification", www.opengeospatial.org/specs, Document 05-005, v1.1, 2005.
- ⁶Berry, Douglas K., *Web Services and Service-Oriented Architecture: The Savvy Manager's Guide*, Morgan-Kaufmann Publishers, San Francisco, 2003.